UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/764,495	01/27/2004	Andrew Mark Nightingale	550-510	6258

23117 7590 01/05/2007
NIXON & VANDERHYE, PC
901 NORTH GLEBE ROAD, 11TH FLOOR
ARLINGTON, VA 22203

EXAMINER	
ALHIJA, SAIF A	
ART UNIT	PAPER NUMBER
2128	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	01/05/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	10/764,495	NIGHTINGALE, ANDREW MARK	
	Examiner	Art Unit	
	Saif A. Alhija	2128	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *18 October 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-51* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-51* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *27 January 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-51 have been presented for examination.

Response to Arguments

2. Applicant's arguments filed 18 October 2006 have been fully considered but they are not persuasive.

i) Applicant argues the 101 rejections of claims 1-51. Applicant has not indicated if the claims are directed to software or hardware. The broadest reasonable interpretation of the claims still defines software per se and as such the rejections are maintained since software per se is not patentable. See MPEP 2106 and 2107. Further, following Applicants amendment claim 35 recites an intended use. The phrase "for use in" in claim 35 is an intended use and therefore the limitations following said phrase are not afforded patentable weight. Finally, the amendments to claim 35 do not fully adopt the Examiners suggestion in the previous office action. See rejection below.

ii) Applicant argues that the reference does not disclose "a debugger operable to control operation of a processing unit associated with the system under verification, the processing unit being operable to execute software routines," in addition to a "debugger signal interface controller." In addition to the citations provided in the previous office action, claim 5 of the reference states:

> A method for design validation as defined in claim 4, wherein the simulation testbench of the individual cores has an event based data format, thereby facilitating a procedure to generate the test pattern signals for debugging of a fault in the cores of the SoC by the verification unit.

This section indicates that the reference generates test pattern signals for debugging a fault which reads on the limitation as recited. Furthermore, Column 9, Lines 16-24 states:

> As the data is design simulation data, a defect free core performs exactly as predicted by the simulation. This response is observed and compared by the control CPU **67** in the verification unit **66**. Any deviation from the simulation is identified by the control CPU **67**. This allows to know the presence of any fault in the core IC on any verification unit (VU) **66**. This step allows to have a fault-free silicon IC of the core on a verification unit (VU) **66** before the SoC level design validation.

This section indicates that the control CPU is configured to detect deviations, which further indicate a fault. Therefore, the reference states a control CPU detecting deviations which are considered faults and thereby allows test pattern signals to be generated to debug the cores. Consequently, the control CPU, for example, controls operation in at least detecting and debugging faults. Accordingly the reference reads on the claims as presented and therefore the rejection is maintained.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

MPEP 2106 recites:

The claimed invention as a whole must accomplish a practical application. That is, it must produce a "useful, concrete and tangible result" State Street 149 F.3d at 1373, 47 USPQ2d at 1601-02. A process that consists solely of the manipulation of an abstract idea is not concrete or tangibles. See In re Warmerdam, 33 F.3d 1354, 1360, 31 USPQ2d 1754, 1759 (Fed.Cir. 1994). See also Schrader, 22 F.3d at 295, 30 USPQ2d at 1459.

3. **Claims 1-51** are rejected because the claimed invention is directed to non-statutory subject matter.

i) Regarding Claims 1-51, the Examiner asserts that the current state of the claim language is such that a reasonable interpretation of the claims would not result in any useful, concrete or tangible result. The Examiner asserts that the claims do not indicate if the methods or apparatus are tangible methods or apparatus utilizing hardware, instead of an arrangement of software lacking tangible embodiment. Further, following Applicants amendment claim 35 recites an intended use. The phrase "for use in" in claim 35 is an intended use and therefore the limitations following said phrase are not afforded patentable weight.

ii) Regarding Claims 35-51, the claim recites a computer program. It should be noted that code (i.e., a computer software program) does not do anything per se. Instead, it is the code stored on a computer that, *when executed*, instructs the computer to perform various functions. The following claim is a generic example of a proper computer program product claim;

A computer program product embodied on a computer-readable medium and comprising code that, when executed, causes a computer to perform the following:

Function A
Function B
Function C, etc...

All Claims dependent upon a rejected a rejected base claim are rejected by virtue of their dependency.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application

filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. **Claims 1-51 are rejected** under 35 U.S.C. 102(e) as being clearly anticipated by **Rajsuman et al. "Method and Apparatus for SOC Design Validation", U.S. Patent No. 6,678,645,** hereafter referred to as **Rajsuman.**

Regarding Claim 1:

Rajsuman discloses Apparatus for performing a sequence of verification tests to perform hardware and software co-verification on a system under verification, comprising:

a plurality of signal interface controllers operable to be coupled to said system under verification, each signal interface controller being operable to perform one or more test actions transferring at least one of one or more stimulus signals and one or more response signals between a corresponding portion of the system under verification and said signal interface controller during performance of said sequence of verification tests; **(Column 5, Lines 43-44. Figure 5)**

a debugger operable to control operation of a processing unit associated with the system under verification, the processing unit being operable to execute software routines; **(Column 9, Lines 16-24. Column 11, Lines 53-63. Figure 5. Claim 5)**

a debugger signal interface controller operable to interface with the debugger and to perform one or more test actions transferring at least one of one or more stimulus signals and one or more response signals between the debugger and said debugger signal interface controller during performance of said sequence of verification tests; **(Column 9, Lines 16-24. Column 11, Lines 53-63. Figure 5. Claim 5)**

and a test manager coupled to said plurality of signal interface controllers and the debugger signal interface controller and operable to transfer test controlling messages to said plurality of signal interface controllers and the debugger signal interface controller identifying the test actions to be performed; **(Column 5, Lines 32-34. Figure 5)**

the test manager being operable to control the operation of the processing unit via the debugger signal interface controller and the debugger in order to co-ordinate the execution of said software routines with the sequence of verification tests. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 2:

Rajsuman discloses Apparatus as claimed in claim 1, further comprising a memory in which the software routines are stored, the debugger signal interface controller being provided with an address indication identifying the addresses of the software routines within the memory, upon receipt of a test controlling message from the test manager, the debugger signal interface controller being operable to generate a corresponding test action with reference to the address indication. **(Column 11, Lines 53-63. Figure 5)**

Regarding Claim 3:

Rajsuman discloses Apparatus as claimed in claim 2, further comprising a status memory operable to store status data, the processing unit being operable to execute monitoring code to monitor the status data in order to identify any changes to the status data and to then execute at least one of said software routines as identified by the change in status data, the corresponding test action being arranged to cause updated status data identifying a corresponding one of said software routines to be stored in the status memory under the control of the debugger. **(Column 11, Lines 53-63. Figure 5)**

Regarding Claim 4:

Rajsuman discloses Apparatus as claimed in claim 3, wherein the debugger is operable to cause the processing unit to store the updated status data in the status memory, whereafter the processing unit reverts to executing the monitoring code. **(Column 11, Lines 53-63. Figure 5)**

Regarding Claim 5:

Rajsuman discloses Apparatus as claimed in claim 2, wherein the processing unit has a plurality of registers associated therewith, and the corresponding test action is arranged to cause data in a selected register of said plurality of registers to be updated under the control of the debugger, such that the processing unit will then execute a corresponding one of said software routines. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 6:

Rajsuman discloses Apparatus as claimed in claim 5, wherein the selected register is operable to store a program counter value and the corresponding test action is arranged to cause the program counter value to be updated in order to cause the processing unit to branch to the corresponding one of said software routines. **(Column 12, Lines 15-35)**

Regarding Claim 7:

Rajsuman discloses Apparatus as claimed in claim 1, wherein upon execution of one of said software routines, the processing unit is operable to update status data indicative of whether the software routine completed successfully, the debugger signal interface controller being operable to perform a predetemmined test action in order to cause a breakpoint to be set by the debugger which is triggered when the processing unit performs said update of the status data. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 8:

Rajsuman discloses Apparatus as claimed in claim 7, wherein the debugger is operable to issue a callback event to the debugger signal interface controller upon triggering of the breakpoint. **(Column 12,**

Lines 15-35)

Regarding Claim 9:

Rajsuman discloses Apparatus as claimed in claim 1, wherein said stimulus and/or response signals are transferred between the debugger signal interface controller and the debugger using Application Programming Interface (API) calls. **(Column 12, Lines 8-14)**

Regarding Claim 10:

Rajsuman discloses Apparatus as claimed in claim 2, wherein at least one of the software routines is written into the memory via the debugger under the control of the debugger signal interface controller. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 11:

Rajsuman discloses Apparatus as claimed in claim 1, wherein multiple processing units are provided, and a corresponding multiple of debugger signal interface controllers are provided, each debugger signal interface controller communicating with the same debugger to cause their respective test actions to be performed. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 12:

Rajsuman discloses Apparatus as claimed in claim 1, wherein the timing of the execution of said software routines is coordinated with the sequence of verification tests. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 13:

Rajsuman discloses Apparatus as claimed in claim 1, wherein the system under verification comprises a plurality of components, each signal interface controller being associated with one of said components. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 14:

Rajsuman discloses Apparatus as claimed in claim 13, wherein the processing unit forms one of the components of the system under verification. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 15:

Rajsuman discloses Apparatus as claimed in claim 1, further comprising the processing unit, the processing unit being provided externally to the system under verification. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 16:

Rajsuman discloses Apparatus as claimed in claim 1, wherein the processing unit comprises a representation of a processor on which the software routines are intended to be executed. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 17:

Rajsuman discloses Apparatus as claimed in claim 1, wherein the system under verification comprises a hardware simulator responsive to said one or more stimulus signals to generate said one or more response signals simulating a response of a data processing apparatus to said one or more stimulus signals if applied to said data processing apparatus. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 18:

Rajsuman discloses A method of performing a sequence of verification tests to perform hardware and software co-verification on a system under verification, comprising the steps of:

performing in each of a plurality of signal interface controllers coupled to said system under verification one or more test actions transferring at least one of one or more stimulus signals and one or more response signals between a corresponding portion of the system under verification and said signal interface controller during performance of said sequence of verification tests; **(Column 5, Lines 43-44. Figure 5)**

controlling via a debugger execution of software routines by a processing unit associated with the system under verification; **(Column 9, Lines 16-24. Column 11, Lines 53-63. Figure 5. Claim 5)**

performing in a debugger signal interface controller coupled with the debugger one or more test actions transferring at least one of one or more stimulus signals and one or more response signals between the debugger and said debugger signal interface controller during performance of said sequence of verification tests; **(Column 9, Lines 16-24. Column 11, Lines 53-63. Figure 5. Claim 5)**

and transferring test controlling messages from a test manager to said plurality of signal interface controllers and to the debugger signal interface controller, the test controlling messages identifying the test actions to be performed; **(Column 5, Lines 32-34. Figure 5)**

whereby the test manager controls the operation of the processing unit via the debugger signal interface controller and the debugger in order to co-ordinate the execution of said software routines with the sequence of verification tests. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 19:

Rajsuman discloses A method as claimed in claim 18, further comprising the steps of:

storing the software routines in a memory; **(Column 11, Lines 53-63. Figure 5)**

providing the debugger signal interface controller with an address indication identifying the addresses of the software routines within the memory;

and upon receipt of a test controlling message from the test manager, generating in the debugger signal interface controller a corresponding test action with reference to the address indication. **(Column 11, Lines 53-63. Figure 5)**

Regarding Claim 20:

Rajsuman discloses A method as claimed in claim 19, further comprising the steps of:

storing status data in a status memory; **(Column 11, Lines 53-63. Figure 5)**

executing on the processing unit monitoring code to monitor the status data in order to identify any changes to the status data and then executing at least one of said software routines as identified by the change in status data, the corresponding test action causing updated status data identifying a corresponding one of said software routines to be stored in the status memory under the control of the debugger. **(Column 11, Lines 53-63. Figure 5)**

Regarding Claim 21:

Rajsuman discloses A method as claimed in claim 20, further comprising the step of: causing the processing unit to store the updated status data in the status memory, whereafter the processing unit reverts to executing the monitoring code. **(Column 11, Lines 53-63. Figure 5)**

Regarding Claim 22:

Rajsuman discloses A method as claimed in claim 19, wherein the processing unit has a plurality of registers associated therewith, and the corresponding test action causes data in a selected register of said

plurality of registers to be updated under the control of the debugger, such that the processing unit will then execute a corresponding one of said software routines. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 23:

Rajsuman discloses A method as claimed in claim 22, wherein the selected register is operable to store a program counter value and the corresponding test action causes the program counter value to be updated in order to cause the processing unit to branch to the corresponding one of said software routines. **(Column 12, Lines 15-35)**

Regarding Claim 24:

Rajsuman discloses A method as claimed in claim 18, further comprising the steps of:

upon execution of one of said software routines, employing the processing unit to update status data indicative of whether the software routine completed successfully; **(Column 5, Lines 41-48. Figure 5)**

and causing the debugger signal interface controller to perform a predetermined test action in order to cause a breakpoint to be set by the debugger which is triggered when the processing unit performs said update of the status data. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 25:

Rajsuman discloses A method as claimed in claim 24, wherein the debugger issues a callback event to the debugger signal interface controller upon triggering of the breakpoint. **(Column 12, Lines 15-35)**

Regarding Claim 26:

Rajsuman discloses A method as claimed in claim 18, wherein said stimulus and/or response signals are transferred between the debugger signal interface controller and the debugger using Application Programming Interface (API) calls. **(Column 12, Lines 8-14)**

Regarding Claim 27:

Rajsuman discloses A method as claimed in claim 19, further comprising the step of: writing at least one of the software routines into the memory via the debugger under the control of the debugger signal interface controller. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 28:

Rajsuman discloses A method as claimed in claim 18, wherein multiple processing units are provided, and a corresponding multiple of debugger signal interface controllers are provided, each debugger signal interface controller communicating with the same debugger to cause their respective test actions to be performed. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 29:

Rajsuman discloses A method as claimed in claim 18, wherein the timing of the execution of said software routines is coordinated with the sequence of verification tests. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 30:

Rajsuman discloses A method as claimed in claim 18, wherein the system under verification comprises a plurality of components, each signal interface controller being associated with one of said components. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 31:

Rajsuman discloses. A method as claimed in claim 30, wherein the processing unit forms one of the components of the system under verification. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 32:

Rajsuman discloses A method as claimed in claim 18, wherein the processing unit is provided externally to the system under verification. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 33:

Rajsuman discloses A method as claimed in claim 18, wherein the processing unit comprises a representation of a processor on which the software routines are intended to be executed. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 34:

Rajsuman discloses A method as claimed in claim 18, wherein the system under verification comprises a hardware simulator which in response to said one or more stimulus signals generates said one or more response signals simulating a response of a data processing apparatus to said one or more stimulus signals if applied to said data processing apparatus. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 35:

Rajsuman discloses A computer program product embodied on a computer-readable medium for use in performing a sequence of verification tests to perform hardware and software co-verification on a system under verification, the computer program product comprising:

a plurality of signal interface controller code blocks operable to be coupled to said system under verification, each signal interface controller code block being when executed causes a computer to perform one or more test actions transferring at least one of one or more stimulus signals and one or more response signals between a corresponding portion of the system under verification and said signal interface controller code block during performance of said sequence of verification tests **(Column 5, Lines 43-44. Figure 5)** ;

debugger code when executed causes the computer to control operation of a processing unit associated with the system under verification, the processing unit being operable to execute software routines; **(Column 9, Lines 16-24. Column 11, Lines 53-63. Figure 5. Claim 5)**

a debugger signal interface controller code block when executed causes the computer to interface with the debugger code and to perform one or more test actions transferring at least one of one or more stimulus signals and one or more response signals between the debugger code and said debugger signal interface controller code block during performance of said sequence of verification tests; **(Column 9, Lines 16-24. Column 11, Lines 53-63. Figure 5. Claim 5)**

and test manager code coupled to said plurality of signal interface controller code blocks and the debugger signal interface controller code block and when executed causes the computer to transfer test controlling messages to said plurality of signal interface controller code blocks and the debugger signal interface controller code block identifying the test actions to be performed; **(Column 5, Lines 32-34. Figure 5)**

the test manager code when executed causes the computer to control the operation of the processing unit via the debugger signal interface controller code block and the debugger code in order to co-ordinate the execution of said software routines with the sequence of verification tests. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 36:

Rajsuman discloses A computer program product as claimed in claim 35, wherein the software routines are stored in a memory, the debugger signal interface controller code block being provided with an address indication identifying the addresses of the software routines within the memory, upon receipt of a test controlling message from the test manager code, the debugger signal interface controller code block when executed causes the computer to generate a corresponding test action with reference to the address indication. **(Column 11, Lines 53-63. Figure 5)**

Regarding Claim 37:

Rajsuman discloses A computer program product as claimed in claim 36, wherein status data is stored in a status memory, the processing unit being operable to execute monitoring code to monitor the status data in order to identify any changes to the status data and to then execute at least one of said software routines as identified by the change in status data, the corresponding test action being arranged to cause updated status data identifying a corresponding one of said software routines to be stored in the status memory under the control of the debugger code. **(Column 11, Lines 53-63. Figure 5)**

Regarding Claim 38:

Rajsuman discloses A computer program product as claimed in claim 37, wherein the debugger code is operable to cause the processing unit to store the updated status data in the status memory, whereafter the processing unit reverts to executing the monitoring code. **(Column 11, Lines 53-63. Figure 5)**

Regarding Claim 39:

Rajsuman discloses A computer program product as claimed in claim 36, wherein the processing unit has a plurality of registers associated therewith, and the corresponding test action is arranged to cause data in a selected register of said plurality of registers to be updated under the control of the debugger code, such that the processing unit will then execute a corresponding one of said software routines. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 40:

Rajsuman discloses A computer program product as claimed in claim 39, wherein the selected register is operable to store a program counter value and the corresponding test action is arranged to cause the program counter value to be updated in order to cause the processing unit to branch to the corresponding one of said software routines. **(Column 12, Lines 15-35)**

Regarding Claim 41:

Rajsuman discloses A computer program product as claimed in claim 35, wherein upon execution of one of said software routines, the processing unit is operable to update status data indicative of whether the software routine completed successfully, the debugger signal interface controller code block when executed causes the computer to perform a predetermined test action in order to cause a breakpoint to be set by the debugger code which is triggered when the processing unit performs said update of the status data. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 42:

Rajsuman discloses A computer program product as claimed in claim 41, wherein the debugger code is operable to issue a callback event to the debugger signal interface controller code block upon triggering of the breakpoint. **(Column 12, Lines 15-35)**

Regarding Claim 43:

Rajsuman discloses A computer program product as claimed in claim 35, wherein said stimulus and/or response signals are transferred between the debugger signal interface controller code block and the debugger code using Application Programming Interface (API) calls. **(Column 12, Lines 8-14)**

Regarding Claim 44:

Rajsuman discloses A computer program product as claimed in claim 36, wherein at least one of the software routines is written into the memory via the debugger code under the control of the computer executing the debugger signal interface controller code block. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 45:

Rajsuman discloses A computer program product as claimed in claim 35, wherein multiple processing units are provided, and a corresponding multiple of debugger signal interface controller code blocks are provided, each debugger signal interface controller code block when executed causes the computer to communicate with the same debugger code to cause their respective test actions to be performed. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 46:

Rajsuman discloses A computer program product as claimed in claim 35, wherein the timing of the execution of said software routines is co-ordinated with the sequence of verification tests. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 47:

Rajsuman discloses A computer program product as claimed in claim 35, wherein the system under verification comprises a plurality of components, each signal interface controller code block being associated with one of said components. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 48:

Rajsuman discloses A computer program product as claimed in claim 47, wherein the processing unit forms one of the components of the system under verification. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 49:

Rajsuman discloses A computer program product as claimed in claim 35, wherein the processing unit is provided externally to the system under verification. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 50:

Rajsuman discloses A computer program product as claimed in claim 35, wherein the processing unit comprises a representation of a processor on which the software routines are intended to be executed. **(Column 5, Lines 41-48. Figure 5)**

Regarding Claim 51:

Rajsuman discloses A computer program product as claimed in claim 35, wherein the system under verification comprises hardware simulator code when executed causes the computer in response to said one or more stimulus signals to generate said one or more response signals simulating a response of a data processing apparatus to said one or more stimulus signals if applied to said data processing apparatus. **(Column 5, Lines 41-48. Figure 5)**

Conclusion

5. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

6. All Claims are rejected.

7. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Saif A. Alhija whose telephone number is (571) 272-8635. The examiner can normally be reached on M-F, 11:00-7:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kamini Shah can be reached on (571) 272-22792279. The fax phone number for the organization where this application or proceeding is assigned is (571) 273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

SAA

December 11, 2006

HUGH JONES PH.D.
PRIMARY PATENT EXAMINER
TECHNOLOGY CENTER 2100